МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное автономное образовательное учреждение высшего образования

«Севастопольский государственный университет»

Кафедра «Информационные системы»

ОТЧЕТ ПО ЛАБОРАТОРНОЙ РАБОТЕ №2

по дисциплине

«Технологии программирования. Объектно-ориентированный подход»

Вариант 8

Выполнил:

Донец Н.О.

Проверил:

Сметанина Т. И.

Севастополь

2024 г.

**1 ЛАБОРАТОРНАЯ РАБОТА №1**

**ИССЛЕДОВАНИЕ ОСОБЕННОСТЕЙ РАБОТЫ ПРОГРАММ С**

**ИСПОЛЬЗОВАНИЕМ КОНТЕЙНЕРОВ. СТАНДАРТНАЯ**

**БИБЛИОТЕКА ШАБЛОНОВ**

**Цель работы:**

Изучить способы реализации и особенности работы программ с

использованием контейнеров стандартной библиотеки шаблонов.

**Задание:**

Написать программу «Работа деканата». В базе хранится информация о

студентах — ФИО, оценки, группа, курс.

Программа должна обеспечивать выбор с помощью меню и выполнение

следующих функций:

– загрузка «базы данных» (из файла) или ее создание;

– общий вывод информации о студентах;

– вывод информации о студентах запрошенной группы, курса;

– вычисление и вывод информации о среднем балле групп.

Хранение данных организовать с применением контейнерного класса

deque.

**Разработка программы:**

Были разработаны класс деканат и студент (Листинг 1).

Листинг 1 – Класс деканата и студента

#ifndef DECANAT

#define DECANAT

#include "Student.h"

#include <deque>

#include <map>

#include <fstream>

using namespace std;

class Decanat {

deque<Student> students;

size\_t studentsAmount = 0;

public:

void AddStudent(Student student);

void LoadStudentsFromFile(string fileName);

void SaveStudentsToFile(string fileName);

void PrintStudents();

void PrintStudents(int cours);

void PrintStudents(string group);

void PrintAVGGradeOfGroups();

};

#endif

#ifndef STUDENT

#define STUDENT

#include <iostream>

#include <map>

using namespace std;

class Student {

public:

string FIO;

map<string, unsigned short> grades;

string group;

unsigned short cours;

// Student(string \_FIO, string \_group, unsigned short \_cours, map<string, unsigned short> \_grades);

void Print();

};

#endif

Также были разработаны методы классов (Листинг 2).

Листинг 2 – Методы деканата и студентов

#include "Decanat.h"

void Decanat::PrintStudents() {

for (auto student : students) {

student.Print();

}

}

void Decanat::PrintStudents(string group) {

for (auto student : students) {

if (student.group == group)

student.Print();

}

}

void Decanat::PrintStudents(int cours) {

for (auto student : students) {

if (student.cours == cours)

student.Print();

}

}

void Decanat::PrintAVGGradeOfGroups() {

map<string, double> groupsGrades;

map<string, size\_t> groupsStudentsAmount;

for (auto student : students) {

for (auto grade : student.grades) {

groupsGrades[student.group] += grade.second;

groupsStudentsAmount[student.group]++;

}

}

for (auto group : groupsGrades) {

cout << "Group: " << group.first << " Average grade: " << group.second/groupsStudentsAmount[group.first] << endl;

}

}

void Decanat::AddStudent(Student student) {

students.push\_back(student);

studentsAmount++;

}

void Decanat::LoadStudentsFromFile(string fileName) {

ifstream file(fileName, ios::binary|ios::in);

Student student;

while(!file.eof()) {

file.read((char\*)&student, sizeof(Student));

AddStudent(student);

}

}

void Decanat::SaveStudentsToFile(string fileName) {

ofstream file(fileName, ios::binary|ios::out);

for (auto student : students) {

file.write((char\*)&student, sizeof(student));

}

}

#include "student.h"

void Student::Print() {

cout << FIO << " Group: "<< group << " Cours: " << cours << endl;

for (auto grade : grades) {

cout << "Subject: " << grade.first << " Grade: " << grade.second << endl;

}

}

Была разработана программа работы с классами (Листинг 3).

Листинг 3 – Работа с классами

#include <iostream>

#include <random>

#include <vector>

#include <string>

#include "Decanat.h"

using namespace std;

std::string GenerateRandomName() {

std::vector<std::string> firstNames = {"Alice", "Bob", "Charlie", "David", "Emma", "Frank"};

std::vector<std::string> lastNames = {"Smith", "Johnson", "Williams", "Jones", "Brown", "Davis"};

std::random\_device rd;

std::mt19937 gen(rd());

std::uniform\_int\_distribution<unsigned short> dist(0, firstNames.size() - 1);

return firstNames[dist(gen)] + " " + lastNames[dist(gen)];

}

std::string GenerateRandomGroup() {

std::vector<std::string> groups = {"IT", "J", "ITIKS", "ABZAC", "CR"};

std::random\_device rd;

std::mt19937 gen(rd());

std::uniform\_int\_distribution<unsigned short> dist(0, groups.size() - 1);

return groups[dist(gen)];

}

Student generateStudent() {

std::random\_device rd;

std::mt19937 gen(rd());

std::uniform\_int\_distribution<unsigned short> dist(1, 10);

Student student;

student.FIO = GenerateRandomName();

student.group = GenerateRandomGroup();

student.cours = dist(gen);

std::vector<std::string> subjects = {"Math", "Physics", "Chemistry", "History", "Literature"};

for (const auto& subject : subjects) {

student.grades[subject] = dist(gen);

}

return student;

}

int main() {

string str;

Decanat decanat;

int number;

int choice = 0;

for (int i = 0; i < 10; i++) {

decanat.AddStudent(generateStudent());

}

do {

cout << "Load from file - 1" << endl;

cout << "Save to file - 2" << endl;

cout << "Print all students - 3" << endl;

cout << "Print students by group - 4" << endl;

cout << "Print students by cours - 5" << endl;

cout << "Print average groups grade - 6" << endl;

cout << "Quit - 0" << endl;

cout<<endl;

cin >> choice;

switch (choice) {

case 1:

cout<<"File: ";

cin>>str;

decanat.LoadStudentsFromFile(str);

break;

case 2:

cout<<"File: ";

cin>>str;

decanat.SaveStudentsToFile(str);

break;

case 3:

decanat.PrintStudents();

break;

case 4:

cout<<"Group: ";

cin>>str;

decanat.PrintStudents(str);

break;

case 5:

cout<<"Cours: ";

cin>>number;

decanat.PrintStudents(number);

break;

case 6:

decanat.PrintAVGGradeOfGroups();

break;

}

} while (choice);

}

Были проведены тесты разработанной программы (рисунки 1 – 3).

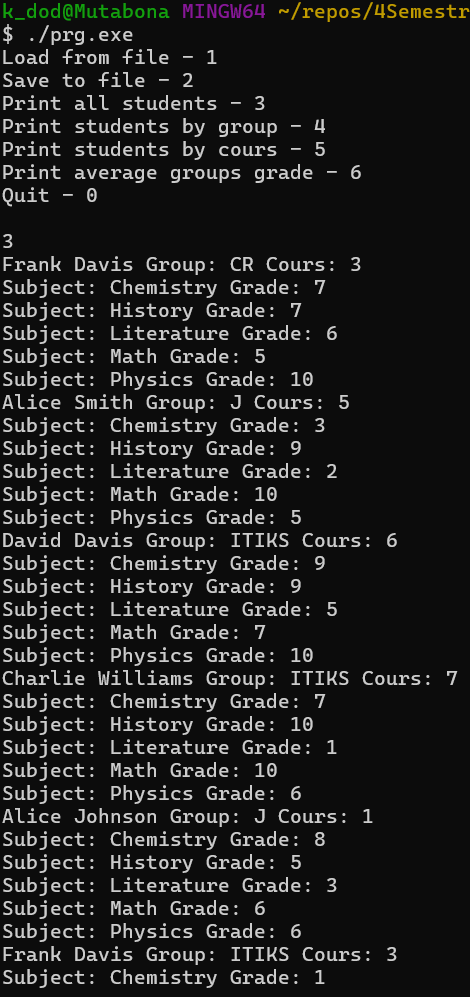


Рисунок 1- Первый тест

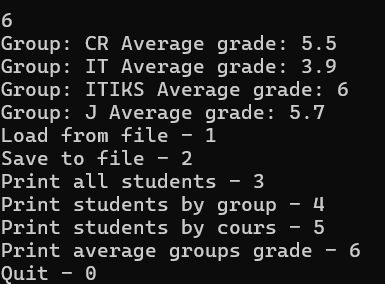


Рисунок 2 – Тест второй

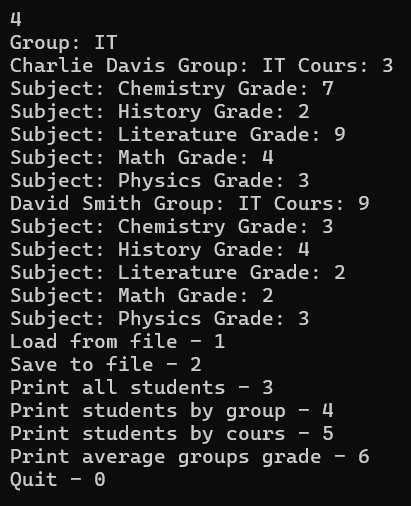


Рисунок 3 – Третий тест

**Вывод:**

В ходе работы были изучены способы реализации и особенности работы программ с использованием контейнеров стандартной библиотеки шаблонов. Также были исследованы основные концепции STL, были рассмотрены понятия контейнеров и их свойств, итераторов, алгоритмов, рассмотрено использование последовательных и ассоциативных контейнеров. Далее была выполнена отладка программы и оформлен отчет по проделанной работе.